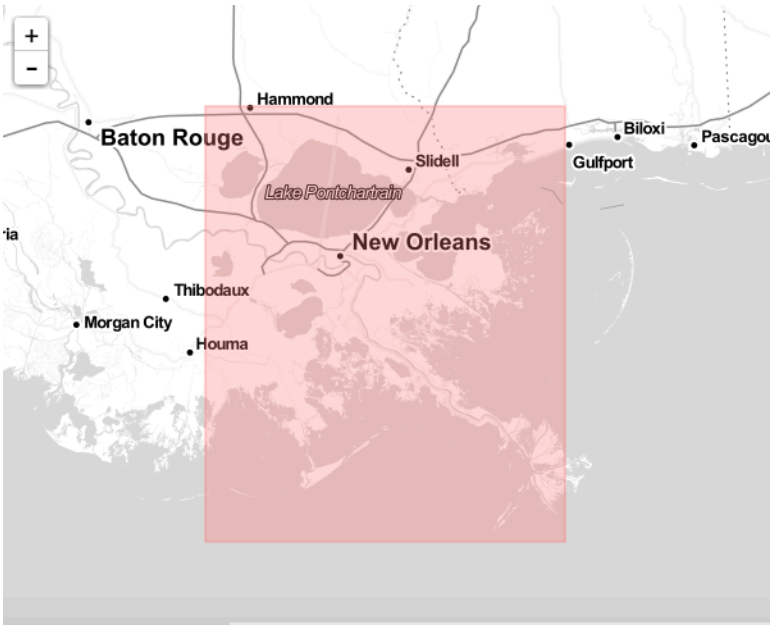
Project 3: OpenStreetMap Data Wrangling with SQL
William Thomas

Map Area



New Orleans, Louisiana

- https://www.openstreetmap.org/relation/131885
- https://mapzen.com/data/metro-extracts/metro/new-orleans_louisiana/

2. Problems Encountered

The problem I encountered in this dataset was the fact that it had a number of street name inconsi stencies and terrible labeling. There were some abbreviations I noticed that were incorrect such a s Royal Street, a well known street in New Orleans, listed as `Royal St,` and needed to be cha nged. I did an audit for postal codes and saw no problems with the data at all.

```
# Function to correct street names using wrong suffix
def update_name(name, mapping):
    """ Substitutes incorrect abbreviation with correct one. """
    m = street_type_re.search(name)
    if m:
        street_type = m.group()

        temp= 0
        try:
            temp = int(street_type)
        except:
            pass

        if street_type not in expected and temp == 0:
            try:
                name = re.sub(street_type_re, mapping[street_type], name)
            except:
                pass
    return name
```

Overview of Data

File sizes

new-orleans_louisiana.osm ..... 1.28 GB
new-orleans_louisiana_sample.osm .. 129.6 MB
Nola.db ..... 66.9 MB
nodes.csv ..... 52.9 MB
nodes_tags.csv ..... 636 KB
ways.csv ..... 2.3 MB
ways_nodes.csv ..... 17 MB
ways_tags.csv ..... 4.9 MB

Number of Ways

query = "SELECT count(*) FROM ways;"
c.execute(query)
c.fetchall()[0][0]

Result:

```
37891
```

Number of Common Way Tags (Top 5)

query = "SELECT key, count(*) FROM ways_tags GROUP BY 1 ORDER BY count(*) DESC LIMIT 5;"
c.execute(query)
c.fetchall()

Result:

```
[(u'building', 22934),
 (u'source', 13851),
 (u'street', 11662),
 (u'housenumber', 11505),
 (u'highway', 5553)]
```

Number of Nodes

query = "SELECT count(*) FROM nodes;"
c.execute(query)
c.fetchall()[0][0]

Result:

```
641497
```

Number of Common Node Tags (Top 5)

query = "SELECT key,count(*) FROM nodes_tags GROUP BY 1 ORDER BY count(*) DESC LIMIT 5;"
c.execute(query)
c.fetchall()

Result:

```
 [(u'housenumber', 9750),
 (u'street', 9750),
 (u'created_by', 3082),
```

```
 (u'power', 1542),
 (u'name', 1202)]
```

Contributors

query = "SELECT temp.user, count(*) as posts FROM (SELECT user, uid FROM ways UNION ALL SELECT user , uid FROM nodes) as temp \
GROUP BY temp.user ORDER BY posts DESC LIMIT 10;"
c.execute(query)
c.fetchall()

Result:

```
[(u'Matt Toups', 344051),
 (u'ELadner', 122690),
 (u'wvdp', 76877),
 (u'coleman_nolaimport', 34880),
 (u'ELadnerImp', 25084),
 (u'woodpeck_fixbot', 21154),
 (u'Matt Toups_nolaimport', 5329),
 (u'Minh Nguyen_nolaimport', 3849),
 (u'ceseifert_nolaimport', 3616),
 (u'Maarten Deen', 2597)]
```

Top Users

query = "SELECT count(DISTINCT(temp.uid)) FROM (SELECT user, uid FROM ways UNION ALL SELECT us er, uid FROM nodes) as temp;"
c.execute(query)
c.fetchall()[0][0]

Result:

```
558
```

Top 10 Amenities

query = "SELECT temp.value, count(*) as num \
FROM (SELECT key,value FROM ways_tags UNION ALL SELECT key,value FROM nodes_tags) as temp \
WHERE temp.key='amenity' GROUP BY temp.value ORDER BY num DESC LIMIT 10;"
c.execute(query)
c.fetchall()

Result:

```
[(u'place_of_worship', 200),
 (u'school', 161),
 (u'restaurant', 62),
```

```
(u'parking', 56),
(u'grave_yard', 51),
(u'kindergarten', 32),
(u'cafe', 26),
(u'fire_station', 26),
(u'bar', 25),
(u'fast_food', 24)]
```

Cuisine

```
query = "SELECT temp.value, count(*) as num \
FROM (SELECT key,value FROM ways_tags UNION ALL SELECT key,value FROM nodes_tags) as temp \
WHERE temp.key='cuisine' GROUP BY temp.value ORDER BY num DESC LIMIT 10;"
c.execute(query)
c.fetchall()
```

Result:

```
[(u'chinese', 6),
(u'regional', 6),
(u'american', 4),
(u'asian', 4),
(u'pizza', 4),
(u'sandwich', 4),
(u'burger', 3),
(u'american;seafood;fish', 2),
(u'coffee_shop', 2),
(u'diner', 2)]
```

Ideas:

The analysis of OpenStreetMap New Orleans has helped me dig into the problems and inconsistency of the OpenStreetMap data. It seems that further cleaning of the data is needed in order to get better and accurate results for analysis. More user contribution would be beneficial to this dataset. Not only having more users would be beneficial, but I would also recommend a universal standard for inputting data in OpenStreetMap data.. A wise move would be to have specific exception handlers that will send out an error if a certain field and or value does not meet the specifications for its type. I believe it would generate efficiency among users who update this data on a daily basis

Anticipated Problems:

A big anticipated problem is lack of awareness about OpenStreetMap. I believe that it is possible that the majority of people and business owners who live in New Orleans, a city known for its culture for festivals and food and less for its technical advancement, know that OSM exist. Other interesting questions would be if do they know that they can add data by themselves? And do they know it is free? Efficiency could also be a problem. Although it isn't required to be a local to contribute to a OpenStreetMap New Orleans dataset or for any OpenStreetMap dataset, getting more people equipped in learning extracting data and learning about SQL could help contribute to improving the data in their respective areas. The reason I brought this point up is because there could be some sort of correlation between users who are from or have lived in the OSM data area they're trying to improve